

JCS20 U.S. PTO
04/19/00

4-21-00
IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

ATTORNEY DOCKET NO. 0544MH-34056 (RM 143)

7
JC586 U.S. PTO
09/551899
04/19/00

In re Application of

HARRI RAJALA et al.

Serial No.: **To Be Assigned**

Filed: **Herewith**

For: **METHOD AND APPARATUS
FOR SUPPORTING MULTIPLE
ALTERNATIVE GRAPHICAL USER
INTERFACES IN COMPUTER-MODERATED
ELECTRONIC COMMERCE**

TRANSMITTAL LETTER

BOX: PATENT APPLICATION
Assistant Commissioner for Patents
Washington, D.C. 20231

Dear Sir:

Enclosed herewith for filing in the above-identified case are:

- Specification containing 31 pages of description, 6 pages of claims and 1 page of abstract;
- Declaration For Patent Application containing a power of attorney;
- 15 sheets of drawings; and
- Our return postcard, which we would appreciate your date stamping and returning to us upon receipt.

"EXPRESS MAIL" NO. EL643425026US

I hereby certify that this paper or fee is being deposited with the United States Postal Service as "Express Mail Post Office to Addressee" service under 37 C.F.R. § 1.10 on the date indicated below and is addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231.

Date of Deposit: 19 April 2000 By: [Signature]

The total filing fee has been calculated as follows:

Basic filing fee = \$690.00

Total filing fee = \$690.00

Please find enclosed our check for \$690.00 to cover the filing.

I hereby authorize the Assistant Commissioner to charge any additional fees which may be required, or credit any overpayment to Deposit Account No. 50-1060. A duplicate copy of this sheet is enclosed.

Respectfully submitted,

Date: 19 April 2000


Kenneth C. Hill
Registration No. 29,650
Melvin A. Hunn
Registration No. 32,574
Hill & Hunn LLP
201 Main Street, Suite 1440
Fort Worth, Texas 76102
(817)332-2113 (voice)
ATTORNEY FOR APPLICANT(S)

Enclosures

cc: Roxanne Morgan, Esq.

SPECIFICATION

Docket No. **0544MH-34056**

TO ALL WHOM IT MAY CONCERN:

BE IT KNOWN that WE, **Harri Rajala, Sami Lahti, Tapani Rautavirta, Samu Lahti, and Markus Salmi**, have invented new and useful improvements in a

**METHOD AND APPARATUS FOR SUPPORTING MULTIPLE ALTERNATIVE
GRAPHICAL USER INTERFACES IN COMPUTER-MODERATED ELECTRONIC
COMMERCE**

of which the following is a specification:

BACKGROUND OF THE INVENTION

1. Field of the Invention:

The present invention relates in general to computer-moderated methods of conducting commercial transactions, and in particular to graphical user interfaces utilized for electronic commerce.

2. Description of the Prior Art:

There has been a sudden increase in the utilization of electronic commerce for the sale of goods and services. This is especially true with the explosion of activity related to Internet commerce, but is also true for commerce conducted utilizing distributed data processing systems which require a dial-in or other similar connection. A great amount of detailed information can be provided to potential buyers over the Internet or a distributed data processing system. This gives the buyer a greater number of choices. And it also gives the seller a greater number of opportunities and options for merchandising its goods or services.

Modern electronic commerce depends on a number of interchanges between potential buyers and sellers through a series of dialog boxes in cascading graphical user interfaces. Such graphical user interfaces include a variety of relatively conventional graphical components which are utilized to communicate the options which are available to the buyer, and to receive the buyer's selection. Typically, the options available to the potential customer are set forth in a manner which allows buttons to be depressed through manipulation of the graphical pointing device. In other words, most of the transaction "dialog" is conducted through the presentation of a series of options to the potential

1 customer and a recordation of the customer's selection of the particular options through
2 the use of a graphical pointing device.

3
4 A variety of relatively well-known companies have relied heavily upon Internet-moderated
5 commercial transactions. One leader in this area is Dell Computer Corporation of Round
6 Rock, Texas. Dell has positioned itself well within the computer hardware market by
7 relying extensively upon the Internet to conduct a relatively extensive dialog with potential
8 customers, all moderated through a series of graphical user interface screens, which
9 present the customer with a variety of options and choices. Through such reliance on the
10 Internet, Dell has been able to keep its overhead low, turn its inventory often, and obtain
11 good results for its shareholders in terms of marketshare growth, increased dividends, and
12 rapid increases in stock value. Dell is just one example of the increasing and highly
13 effective use of the Internet to conduct commerce.

14
15 For those engaged in electronic commerce, any means which can be utilized to further
16 reduce costs and to stabilize operations will likely be appreciated and adopted.
17

SUMMARY OF THE INVENTION

It is one objective of the present invention to provide an improved method and apparatus for supporting multiple alternative graphical user interfaces in computer-moderated electronic commerce.

It is another objective of the present invention to provide an improved method and apparatus for supporting a low bandwidth graphical user interface for use in computer-moderated electronic commerce.

It is a particular objective of the present invention to provide an improved method and apparatus for supporting a low bandwidth HTML graphical user interface for use in computer-moderated electronic commerce.

These and other objectives are achieved as is now described.

BRIEF DESCRIPTION OF THE DRAWINGS

The novel features of the present invention are set forth in the appended claims. The invention itself, however, as well as a preferred mode of use, further objectives and advantages thereof, will best be understood by reference to the following detailed description of the preferred embodiment when read in conjunction with the accompanying drawings, wherein:

Figure 1 is a simplified and pictorial representation of a distributed data processing system;

Figure 2 is a simplified block diagram representation of an inside sales business object server;

Figure 3 is a block diagram and pictorial representation of the scalability of the present invention which represents one commercially advantageous product feature;

Figure 4 is a block diagram and pictorial representation of the multithread capability of the computer moderated electronic commerce system of the present invention which represents another commercially advantageous feature;

Figure 5 is a block diagram and pictorial representation of an overview of the user interface options available utilizing the present invention which represents still another commercially advantageous product feature;

Figure 6 is a pictorial representation of an exemplary rule maintenance module;

Figure 7 is a pictorial representation of an exemplary dialog control screen in accordance with the present invention;

Figure 8 is a pictorial representation of an exemplary dialog manager screen in accordance with the present invention;

Figure 9 is a pictorial representation of an exemplary user interface maintenance operation in accordance with the present invention;

Figure 10 is an exemplary pictorial and block diagram representation of a plurality of user interfaces which may be supported utilizing the present invention;

Figure 11 is a flowchart representation of the utilization of a rendering engine to dynamically construct a series of graphical user interfaces, in real time, and utilizing a single database;

Figure 12 is a block diagram representation of the utilization of the present invention to support multiple alternative rendering engines from a single database which maintains metadata relating to the objects of the commercial transaction;

Figure 13 is a table of values used in conjunction with the preferred embodiment;

Figure 14 is an example illustrating element layout in an HTML environment;

1 **Figure 15** depicts a populated HTML page in accordance with a preferred embodiment of
2 the present invention; and

3
4 **Figure 16** is a flowchart illustrating an implementation of the preferred embodiment.

5

Continued on next page

DETAILED DESCRIPTION OF THE INVENTION

With reference now to the figures and in particular with reference to **Figure 1**, there is depicted a pictorial representation of the Internet 106 and a distributed data processing system 108 either or both of which may be utilized to implement the method and system of the present invention. As may be seen, distributed data processing system 108 may include a plurality of networks, such as Local Area Networks (LAN) 110 and 132, each of which preferably includes a plurality of individual computers 112 and 130, respectively. Of course, those skilled in the art will appreciate that a plurality of Intelligent Work Stations (IWS) coupled to a host processor may be utilized for each such network.

As is common in such data processing systems, each individual computer may be coupled to a storage device 114 and/or a printer/output device 116. One or more such storage devices 114 may be utilized, to store the various data objects or documents which may be periodically accessed and processed by a user within distributed data processing system 108.

Still referring to **Figure 1**, it may be seen that distributed data processing system 108 may also include multiple mainframe computers, such as mainframe computer 118, which may be preferably coupled to Local Area Network (LAN) 110 by means of communications link 122. Mainframe computer 118 may also be coupled to a storage device 120 which may serve as remote storage for Local Area Network (LAN) 110. A second Local Area Network (LAN) 132 may be coupled to Local Area Network (LAN) 110 via communications controller 126 and communications link 134 to a gateway server 128. Gateway server 128 is preferably an individual computer or Intelligent Work Station (IWS) which serves to link Local Area Network (LAN) 132 to Local Area Network (LAN) 110.

1 As discussed above with respect to Local Area Network (LAN) 132 and Local Area
2 Network (LAN) 110, a plurality of data processing procedures or documents may be stored
3 within storage device 120 and controlled by mainframe computer 118, as Resource
4 Manager or Library Service for the data processing procedures and documents thus
5 stored.

6
7 Of course, those skilled in the art will appreciate that mainframe computer 118 may be
8 located a great geographical distance from Local Area Network (LAN) 110 and similarly
9 Local Area Network (LAN) 110 may be located a substantial distance from Local Area
10 Network (LAN) 132. That is, for example, Local Area Network (LAN) 132 may be located
11 in California while Local Area Network (LAN) 110 may be located within Texas and
12 mainframe computer 118 may be located in New York.

13
14 **Figure 1** also shows a graphical representation of a connection between local area
15 network 110 and the Internet 106 (World Wide Web) which is a wide area network
16 connecting thousands of disparate networks in industry, education, government, and
17 research, which utilizes TCP/IP as the standard for transmitting information. As is shown
18 in the view of **Figure 1**, a plurality of buyers (B1, B2, B3) may connect through the Internet
19 106 to local area network 110 which, for purposes of explanation, represents a local area
20 network under the control of a seller conducting electronic commerce. Alternatively, and
21 simultaneously, a plurality of buyers (B4, B5, B6) may make a dial-in connection through
22 gateway server 128 to local area network 110 in order to conduct electronic commerce
23 with the seller which maintains control over local area network 110. The seller may utilize
24 either or both the Internet connection or dial-in connection in order to communicate
25 simultaneously with a number of buyers, including B1, B2, B3, B4, B5, and/or B6.

Figure 2 is a simplified block diagram representation of an inside sales business object server 202 which is connected via read/write bus 206 to sales data server 204. Sales business object server 202 is composed of a number of software packages which are dedicated to particular electronic commerce functions. A read/write bus 206 allows sales business object server 202 to read data from sales data server 204, and also allows sales business object server 202 to write data to the sales data server 204. The sales business object server 202 includes a suite of server services modules 208 and a suite of data model modules 210. Preferably, the server services modules 208 include a session management module 212, a product data module 214, a document generator module 216, a pricing engine module 218, a configuration engine module 220, and a quote/shopping cart module 222. The session management module 212 is utilized to control the electronic dialog between each buyer or potential buyer and the seller and typically includes product names, product serial numbers, product descriptions, and product fact sheets as well as other similar information. The product data module 214 includes product information about the product array offered by the seller. The document generator module 216 is utilized to generate printed materials which support the electronic transaction, such as shipping and invoice documentation. The pricing engine module 218 is utilized to allow the seller to intelligently and accurately generate the total or "end" price to the customer. Preferably, the pricing engine module will take into account differences in currency, various discounts offered by the seller, the cost of freight and shipping, and the cost of warranties and extended service options. The configuration engine module 220 is utilized in order to at least partially automate a process through which the customer's needs are analyzed. Additionally, the configuration engine module is utilized to determine component compatibility in order to prevent the attempted combination of parts, components, or assemblies which are incompatible or not optimal. The quote/shopping cart module 222 is utilized to generate quotation documentation (both electronic and paper documents), if

1 necessary, and which builds automatically a quotation based upon configuration
2 information, price information, and quote information.

3
4 The data model modules 210 includes a plurality of software modules including a quote
5 identification module 224, a quote version module 226, a configuration module 228, and a
6 customer module 230. The quote identification module 224 is utilized to keep track of the
7 identification of various quotes made to potential and actual customers. The quote version
8 module 226 allows the system to keep track of a series of quotations to potential
9 customers and actual customers in order to prevent inconsistencies between quotes, or
10 other losses of information. The configuration module 228 maintains any information
11 relating to particular potential or proposed configurations. The customer module 230
12 maintains customer information such as identity, address, telephone, payment method,
13 and similar information.

14
15 Collectively, the server services modules 208 and data model modules 210 cooperate to
16 allow orderly and efficient communication between the seller and potential buyers through
17 sales data server 204.

18
19 Some advantages that may be obtained utilizing the present invention are graphically
20 depicted in **Figures 3** and **4**. **Figure 3** is a block diagram and pictorial representation of
21 the scalability of the present invention which represents one commercially advantageous
22 product feature, while **Figure 4** is a block diagram and pictorial representation of the
23 multithread capability of the computer moderated electronic commerce system of the
24 present invention which represents another commercially advantageous feature. As
25 depicted in **Figure 3**, a database server 240 may support a plurality of sales data servers,
26 including sales data servers 204, 242. Each of the sales data servers 204, 242 may in

1 turn support a number of sales business object servers. For example, sales data server
2 204 may support sales business object servers 244, 246, 248; additionally, sales data
3 server 242 may support sales business object servers 250, 252, 254. In this manner, the
4 computer moderated electronic commerce system of the present invention can be scaled
5 in a pyramidal fashion which allows for a theoretically unlimited number of concurrent
6 users being supported.

7
8 The computer moderated electronic commerce system of the present invention also allows
9 simultaneous multithread operation, as is depicted in block diagram in pictorial form in
10 **Figure 4**. As is shown, a data server 260 may be utilized to allow database connection
11 pooling. This optimizes the efficiency of database resource utilization. In the preferred
12 embodiment, an oracle database may be utilized. The data server 260 provides the data
13 to the application server layer which is composed of a number of separate components,
14 some of which were discussed above in connection with **Figure 2**. As is shown, sales
15 data server 204 is communicatively and operationally coupled with sales business object
16 server 202 which is capable of supporting a number of simultaneous electronic commerce
17 communication sessions including sessions 270, 272, 274, 276. The sales business
18 object server 202 is communicatively and operationally coupled to sales internet server
19 280 which is in turn communicatively and operationally coupled to web server 282. Web
20 server 282 allows for the simultaneous communication with potential buyers at and utilizing
21 computing devices 290, 292, 294, 296, and 298. Essentially, sales business object server
22 202 is a multithread server which utilizes multiprocessor servers. Multiple user sessions
23 may be enabled in one process, which makes for efficient use of the operating system
24 resources. The utilization of web server 282 and Internet sales server 280 facilitates
25 relatively low-cost simultaneous communication with buyers and potential buyers. The
26 communication between web server 282 and the computers (290, 292, 294, 296, and 298)

1 which are under control of the buyers and potential buyers is conducted utilizing a
2 relatively low bandwidth HTML dialog.

3
4 In the prior art, companies that conducted electronic commerce did so utilizing a Windows-
5 based graphical user interface which was accessed through conventional dial-in
6 operations. The Windows-based graphical user interface typically communicated with a
7 tabular database (such as a Sequel database). The bandwidth for communication
8 between the Windows-based graphical user interface and the tabular database was
9 relatively large in comparison with the bandwidth available for HTML-moderated
10 communications via the Internet. As a practical manner, many entities that currently
11 conduct electronic commerce have significant investment in non-Internet facilities including
12 dial-up capabilities and Windows-based graphical user interfaces, as well as substantial
13 investments in databases relating to the products, pricing, and customers. As electronic
14 commerce migrates from dial-up sessions (where bandwidth presented no real problems)
15 toward Internet-based communications (where bandwidth becomes a substantial problem)
16 a significant number of problems must be resolved. Preferably, and in accordance with
17 the present invention, an entity's investment in its electronic commerce systems need not
18 be wasted. In accordance with the present invention, a variety of graphical user interfaces
19 may be constructed and maintained, without requiring reinvestment in, or recreation of, the
20 databases which supported such electronic commerce.

21
22 **Figure 5** is a block diagram and pictorial representation of an overview of the interface
23 options available utilizing the present invention which represents still another commercially
24 advantageous product feature. The view of **Figure 5** represents the different layers of
25 processing which must be implemented in order to provide the system of the present
26 invention. As is shown, the data server 260 maintains the metadata which is utilized

1 during the commercial transaction. This may include component identification and
2 component property of the various items offered for sale by the seller. The data server
3 260 communicates with the sales data server 204. The sales data server 204
4 communicates with the sales business object server 202. The sales business object
5 server 202 may communicate over communication link 303 with data processing system
6 296. The communication between the sales business object server 202 and data
7 processing system 296 may be conducted utilizing the JAVA programming language;
8 therefore, the client-user interface is supported by JAVA and communication is conducted
9 over a local area network or a high speed wide area network.

10
11 An alternative means of communicating with buyers is also provided. As is shown in
12 **Figure 5**, sales internet server 280 may also communicate with sales business object
13 server 202. Web server 282 in turn communicates with sales internet server 280. Web
14 server 282 communicates intermittently over communication link 300 utilizing an HTTP
15 protocol. Communication is conducted with one or more data processing systems such as
16 data processing system 290. Data processing system 290 is operating in a Web browser
17 mode of operation. Communication between data processing system 290 and Web server
18 282 is conducted utilizing HTML. The communication link 300 is a low-speed wide-area
19 network or internet connection. The view of
20

21 **Figure 5** shows two different user interfaces which are supported by the data maintained
22 in data server 260. The dialog which makes up the content of the electronic transaction is
23 constructed dynamically by the sales data server 204 and sales business object server
24 202 utilizing the metadata maintained on data server 260. In this way, a single data base
25 may be utilized to support a plurality of simultaneous transaction dialogs in a plurality of

1 differing programming interfaces supported by different programming languages and over
2 communication links with different capabilities.

3 **Figure 6** is a pictorial representation of an exemplary rule maintenance module.
4 The content of the dialog may be maintained utilizing such a rule maintenance system
5 which allows for the visual mapping of information. In this manner, the various
6 components, sub-assemblies, and buyer options may be mapped out in a logical manner.
7 The dialog between the seller and potential buyers may be constructed utilizing a "Dialog
8 Control" module, which is depicted in **Figure 7**, and maybe manually generated utilizing a
9 "Dialog Manager" module which is depicted in **Figure 8**. As is shown, the Dialog Control
10 module in **Figure 7** is utilized to enumerate the various dialog controls which are
11 associated with a particular graphical interface screen. The example depicted in **Figure 7**
12 relates to a dialog I.D. relating to the "chassis" of a vehicle. The seller may utilize the
13 various command buttons on the right side of the screen to, add, delete, edit, clear the
14 various controls which can be, or are, in fact, associated with a particular graphic user
15 interface. The Dialog Manager in **Figure 8** depicts the dialog associated with a "cabin"
16 graphical user interface. As is shown, the dialog includes ListBox1, ListBox3, CheckBox1,
17 ListBox4, ListBox2, TextBox1, CheckBox2, and PushButton1. Each one of these
18 graphical user interface items has various "property" and "value" attributes associated
19 therewith. **Figure 9** depicts a number of simultaneously open and overlapping screens,
20 including a Sales Maintenance screen which maps the relationship between the
21 components, a Dialog Manager screen which lists the various dialog items associated with
22 the "cabin" graphical user interface, and a Dialog Editor screen which is utilized to edit the
23 Dialog Manager screen. Utilizing these functions, all of which are known in the prior art,
24 one may construct graphical user interface screens which are logically linked to one
25 another, which include a variety of dialog items, and which may be edited when product
26 content changes.

1
2 **Figure 10** is a pictorial and graphic representation of the maintenance of the various user
3 interfaces in accordance with the preferred embodiment of the present invention. As is
4 shown, a product rule set 310 is utilized to map or organize the various product features
5 and options which are offered by the seller to potential buyers. The Dialog Controls
6 module 312 is utilized to control the content of the dialog boxes of a graphical user
7 interface. The content may be edited utilizing Dialog Editor 314. In accordance with the
8 preferred embodiment of the present invention database 260 maintains the metadata
9 associated with the product line and options which are set forth in the product rule set 310.
10 Database 260 supports a variety of rendering engines which are utilized to generate
11 graphical user interface screens as part of the dialog transaction. As is shown, JAVA-
12 rendering engine 320 accesses database 260 in order to generate dynamically and in real
13 time user interface 326 which is a JAVA applet. HTML-rendering engine 322 utilizes
14 database 260 to generate dynamically and in real time graphical user interface screen 328
15 which is an HTML user interface. C++-rendering engine 324 utilizes data from database
16 260 to generate dynamically and in real time graphical user interface 330. Note that
17 graphical user interfaces 326, 328, and 330 are all very similar in their content and layout,
18 even though they are generated in different operating environments utilizing different
19 programming languages. All of the rendering engines 220, 222, and 224, make dynamic
20 use of the Dialog Control box 312 in order to generate the graphical user interfaces 326,
21 328, and 333.

22
23 A method and apparatus is provided which allows for a single database to support
24 dynamically a number of multiple alternative graphical user interfaces, all in order to allow
25 computer moderated electronic commerce. The database interacts with a number of
26 rendering engines which utilize metadata maintained in the database to dynamically

1 generate an electronic commerce transaction dialog in a plurality of different
2 communication, operation, and programming environments. For example, the present
3 invention may be utilized to simultaneously support graphical user interfaces constructed
4 utilizing the programming language C++. Alternatively, and additionally, the system may
5 support a graphical user interface constructed utilizing the JAVA programming language.
6 Alternatively, and additionally, the invention allows a single database to support a relatively
7 low band with communication in an internet environment utilizing the HTML programming
8 language. In accordance with the present invention, the differing graphical user interfaces
9 may have the same general appearance and the same series of dialog boxes in order to
10 enable the electronic commerce.

11
12 **Figure 11** is a flowchart representation of the utilization of a rendering engine to
13 dynamically construct a series of graphical user interfaces, in real time, and utilizing a
14 single database. The process starts at block 402, and continues at block 404, wherein
15 communication is established between a buyer and a seller. This communication will
16 occur over a particular communication channel. For example, the buyer may make contact
17 with the seller utilizing the Internet by accessing the seller's website. Alternatively, the
18 buyer may dial-in to a wide area network utilizing conventional telecommunications
19 modem connections in order to communicate with the seller utilizing a graphical user
20 interface constructed in the conventional manner utilizing the C++ programming language.
21 Alternatively, the potential buyer may make communication with the seller utilizing JAVA
22 applets. All three of these scenarios are alternatives to one another and are graphically
23 depicted in **Figure 10**. In accordance with step 406, the seller's data processing system
24 activates the appropriate rendering engine which is suited for the mode and channel of
25 communication which has been established by the buyer. For example, the buyer is
26 making contact utilizing the Internet, the HTML rendering engine will be activated. Then,

1 in accordance with step 408, the seller's data processing system calls the appropriate
2 dialog manager module. In actual practice, an electronic transaction is composed of a
3 series of cascading and logically-linked graphical user interfaces. Each graphical user
4 interface has associated with it a particular dialog manager module, as is conventional.
5 The dialog manager module identifies each and every component of the graphical user
6 interface which will be presented to the buyer. Some of the components are "inactive"
7 components and merely present images, data, or information; however, other elements of
8 the graphical user interface are "active" elements which are adapted to receive user input
9 typically through the detection of the operator actuation of the graphical pointing device
10 (typically the depression of the left button on the mouse associated with the buyer's
11 computer). Then, in accordance with step 410, the seller's data processing system utilizes
12 the rendering engine to generate an associated graphical user interface. Concurrently
13 with this step, and as is set forth in step 412, the data processing system of the seller
14 communicates with a single database in order to read metadata which is associated with
15 the graphical user interface. As described above, the metadata may be arranged utilizing
16 conventional tools such as a rule maintenance module. The metadata may comprise
17 simple product number and feature information; however, in alternative electronic
18 transactions, the metadata may comprise a substantial body of transaction and product
19 information. The more complex the subject matter of the electronic transactions, the more
20 likely there is to be associated with each graphical user interface a greater amount of
21 detailed information. This communication between the rendering engine and the single
22 database is conducted in real time and is done so dynamically during the interaction
23 between the seller's data processing system and the buyer's data processing system.
24 One significant advantage of this approach is that the seller need not maintain multiple
25 parallel databases for each rendering engine; instead, a single database may be
26 maintained. This is a low cost option since there are greater costs associated with

1 maintaining several parallel databases and it is often difficult to maintain consistency
2 between such databases.

3
4 In accordance with step 414, the data processing system under the control of the seller
5 applies the metadata dynamically and in real time during creation of the graphical user
6 interface. Then, in accordance with block 416, the seller's data processing system
7 monitors for the dialog in order to determine input or selection of options by the buyer. In
8 accordance with step 418, if necessary, the data processing system under the control of
9 the seller is utilized to write data to the database. Then, in accordance with block 420, the
10 data processing system which is under the control of the seller monitors for a termination
11 of the communication session. If the session is ended, the process ends at block 422.
12 However, if the session is not ended, the data processing system under control of the
13 seller monitors the dialog as conducted through the dialog boxes and returns to block 408,
14 wherein the appropriate dialog monitor module is called based upon the buyer's input. The
15 process repeats over and over again until the dialog is terminated. As a consequence of
16 the dialog, a substantial amount of detailed metadata is sequentially presented to the
17 potential buyer in a series of cascading graphical user interface dialog boxes. In turn, the
18 data and selections provided by the buyer may be recorded to the single database in order
19 to enable completion of the transaction or return to the transaction at a later date. It is
20 widely known that many electronic interactions do not result in a transaction at the first
21 contact. It may take several interactions with the buyer before a transaction is completed.
22 This is the reason that most electronic transaction systems have shopping carts which
23 may be preserved in memory and recalled at a later date by the buyer in order to allow the
24 buyer to modify or add to the shopping cart.

Figure 12 is a block diagram representation of the utilization of the present invention to support multiple alternative rendering engines from a single database which maintains metadata relating to the objects of the commercial transaction. As is shown in **Figure 12**, the buyer system 500 communicates with a data processing system under control of the seller through communication channel 504. The communication channel may be a dial-in modem connection or an Internet connection. The mode of accessing the seller's data processing system will determine which particular one of a number of available alternative rendering engines are employed by the seller's data processing system. As is shown in **Figure 12**, rendering engines 506, 508, 510 may be utilized to support a variety of alternative communication modes. For example, if the various rendering engines may support communication through JAVA scripts, HTML protocols, or through a graphical user interface constructed utilizing C++. Alternatively, other different programming languages and communication modes may be supported by different rendering engines. Once the appropriate rendering engine is selected, it communicates with a dialog manager module 512. The dialog manager module communicates through rule maintenance module 518 to database 520 which contains product metadata. The dialog manager modules correspond to various graphical user interfaces which are provide to the buyer's system 500 during a series of dialog box communications. The dialog manager module 512 is constructed utilizing a dialog editor 516 through available dialog controls 514. The rendering engine 506 may pass information to database 512 for recordation.

Alternatively and/or concurrently, a number of buyers including buyers 502 and 522 may communicate through communication channels 524, 526 with alternative rendering engines including rendering engines 508, 510. These electronic transactions are communicated through a series of graphical user interfaces which present a series of dialog boxes to the buyer's system 502, 522 and which receive the input of the buyer and

1 selections of the buyer through monitoring the utilization of the graphical pointing device.
2 Rendering engines 508, 510 also access database 520 through dialog manager 512 and
3 rule maintenance module 518. In this manner, a single database 520 is utilized to support
4 a plurality of alternative rendering engines 506, 508, 510, which allow for simultaneous
5 concurrent communication over different communication channels 504, 524, 526 with a
6 plurality of buyer data processing systems 500, 502, 522.

7
8 The dialog editor 516 of **Figure 12** is similar to a dialog editor utilized for creating and
9 editing dialogs for the “rhythm” product of i2 Technologies, assignee of the present
10 application. It includes many editing functions which allow a dialog to be either created or
11 edited. Such functions include the following specific functions:

- 12 • NEW: This command allows the programmer to create a new dialog box.
- 13 • OPEN: This command allows the programmer to open an existing dialog
14 box template.
- 15 • SAVE: This command allows a programmer to create a file for the current
16 dialog box template and save that template to the file.
- 17 • SAVE AS: This command allows the programmer to save the current dialog
18 box template in a file under a new name.
- 19 • TEST DIALOG: This command allows the programmer to toggle between a
20 run mode in which the dialog box “comes alive” for testing purposes in an
21 edit mode in which one can make changes to the dialog box.
- 22 • CAPTURE DIALOG: This command allows the programmer to capture
23 standard window controls from a standard window dialog box in another
24 windows application.
- 25 • EXIT: This command allows a programmer to close the dialog editor
26 module.

- 1 • EXIT AND RETURN: This command allows a programmer to close a dialog
- 2 editor and return to the host application.
- 3 • EDIT MENU: This command allows a programmer to undue up to ten
- 4 preceding operations.
- 5 • CUT: This command allows the programmer to select a dialog box control
- 6 from an application window and place it on a clipboard.
- 7 • COPY: This command allows the programmer to copy a selected dialog box
- 8 control without removing it from the dialog editor's application window and
- 9 places it on a clipboard.
- 10 • PASTE: This command allows the programmer to insert the contents of the
- 11 clipboard into a dialog editor.
- 12 • CLEAR: This command removes a selected dialog box control from the
- 13 programmer's application window without placing it on the clipboard.
- 14 • DUPLICATE: This command allows a programmer to create a duplicate
- 15 copy of a selected control.
- 16 • SIZE TO TEXT: This command allows a programmer to adjust the board as
- 17 to certain controls to fit the text displayed on them.
- 18 • GRID: This command allows a programmer to display a grid dialog box
- 19 which can be used to display or turn off the grid and adjust the grid spacing.

20 The Dialog Editor 516 further includes a number of commands which are utilized to add
21 control or communication elements to a particular user interface. Some principal control
22 menu items include the following:

- 23 • OK BUTTON: This command adds a default "OK" button to a dialog box.
- 24 • CANCEL BUTTON: This command allows the programmer to add a default
- 25 cancel button to a dialog box.

- 1 • HELP BUTTON: This command allows the programmer to add a help
2 button to a dialog box.
- 3 • PUSH BUTTON: This command allows the programmer to add a push or
4 command button to a dialog box.
- 5 • OPTION BUTTON: This command allows the programmer to add an option
6 button to a dialog box. An option button is one of two or more link buttons
7 that let user select only one from a group of mutually exclusive choices.
- 8 • CHECK BOX: This command allows the programmer to add a check box to
9 a dialog box. Users can check or clear a check box to indicate their
10 preference regarding the alternatives specified on the check box label.
- 11 • GROUP BOX: This adds a group box to a dialog box. A group box is a
12 rectangular design element used to enclose a group of related controls.
13 One can use the optional group box label to display a title for the controls in
14 the box.
- 15 • TEXT: This command allows a programmer to add a text control to a dialog
16 box. A text control is a field containing text you want to display for the user's
17 information. The text in this field wraps, and the field can contain a
18 maximum of 255 characters. Text controls can either display stand-alone
19 text or be used as labels for text boxes, list boxes, combo boxes, drop list
20 boxes, pictures, and picture buttons. One can choose the font in which the
21 text appears.
- 22 • TEXT BOX: This control allows the addition of a text box to a dialog box. A
23 text box is a field in which users can enter text (potentially, as much as 32K).
24 By default, this field holds a single line of non-wrapping text. If one chooses
25 a multi-line setting in the text box,, this field will hold multiple lines of
26 wrapping text.

- 1 • LIST BOX: This adds a list box to a dialog box. A list box is a displayed,
2 scrollable list from which the user can select one item.
- 3 • COMBO BOX: This adds a combo box to a dialog box. A combo box
4 consists of a text field with a displayed, scrollable list beneath it. Users can
5 either select an item from the list or enter the name of the desired item in the
6 field text. The currently selected item is displayed in the field text. If the item
7 was selected from the scrolling list, it is highlighted there as well.
- 8 • DROP LIST BOX: This adds a drop list box to a dialog box. A drop list box
9 consists of a field that displays the current selected item, followed by a
10 downward-pointing error which users can click to temporarily display a
11 scrolling list of items. Once they select an item from the list, the list
12 disappears and the newly selected item is displayed in the field.
- 13 • PICTURE: This command adds a picture to a dialog box. A picture is a field
14 used to display a windows bit map or metafile.
- 15 • PICTURE BUTTON: This adds a picture button to a dialog box. A picture
16 box is a special type of push, or command button on which a windows bit
17 map or metafile appears.
- 18 • PICT TOOL: This enables a programming to select, move, and resize, items
19 that control the insert point.

20 The dialog editor of 516 can be considered to contain a pallet of functions, commands,
21 and graphical elements which may be selected by a programmer in establishing the
22 contents of a dialog which consists of one or more cascading or linked graphical user
23 interfaces. The programmer utilizes the elements in the dialog editor in order to construct
24 and edit a dialog.

1 The dialog controls module 514 is comparable to a dialog controls module present in the
2 "rhythm" product sold by i2 Technologies, Inc. Assignee of the present application. This
3 module allows one to connect structure boxes to dialog controls one has created utilizing
4 the dialog editor 516. An exemplary screen from the dialog controls module 514 of **Figure**
5 **12** is depicted in **Figure 7**. As is shown, a "NAME 601" field is provided which displays the
6 name of the active structure box. Additionally, a "CODE FIELD 603" is provided which
7 displays the code of a selected box. One can select another code from a drop down list.
8 The boxes are listed in the same order as they appear in the structure. When a particular
9 code is identified in Code Box 603, a grid 605 is displayed there below which displays all
10 dialogs, controls and variables that have been earlier designated as being associated with
11 the box or boxes. As shown in **Figure 3**, a grid 605 includes a variable field 607, a dialog
12 ID field 609 and a control field 611.

13
14 The same screen depicted in **Figure 7** also depicts a series of fields which identify dialogs.
15 A dialog name Box 613 is provided which identifies a particular dialog. A dialog ID box
16 615 displays a dialog ID that one has entered for a dialog in the dialog manager. A dialog
17 control box 617 displays the dialog controls that have been created utilizing the dialog
18 editor 516 of **Figure 12**.

19
20 As is shown in **Figure 7**, a variety of control buttons are provided including an "OK" button,
21 a "cancel button", "scrolling" buttons, a "variables" button which opens a variables dialog, a
22 "clear" button which clears the PM variable from the selected control for the active box. An
23 "add" button which can be utilized to add controls. A "delete" button which removes
24 selected control from the active box, an "update" button which updates the active dialog
25 and control as new ones in the grid, an "edit" button which opens the dialog manager

1 dialog, a “show” button which displays the dialog with the box codes attached to the
2 controls, and a “close” button which closes the show dialog.

3
4 An exemplary screen from the dialog manager 512 of **Figure 12** is depicted in the view of
5 **Figure 8**. This dialog manager is similar to the dialog manager present in the
6 “RHYTHMS” product of i2 Technologies which is the Assignee of the present application.
7 The exemplary dialog manager screen includes an ID field 651 which displays the ID of
8 the active dialog. A drop down list may be provided to allow the selection of another
9 dialog. It further includes a name field 653 which displays the name of the active dialog. It
10 contains a controls field which displays a list of controls related to the active dialog. The
11 controls are buttons, radio buttons, group boxes, etc.

12
13 A properties grid 657 is also provided which contains the values from the dialog editor
14 established for graphical user interface elements. A plurality of control buttons are also
15 provided in the view of **Figure 8**. A programmer may create a new dialog by clicking the
16 “ADD” button. The programmer may create a new dialog by clicking the “DLG EDITOR”
17 button. The scrolling buttons are provided to allow backwards and forwards scrolling. A
18 “SAVE” button is provided to allow the saving of a edited or newly created dialog. The
19 “DELETE” button allows the deletion of an active dialog.

20
21 **Figure 13** is a table which identifies the primary values which are under the parameters
22 control. A “CLASS” value cannot be edited. However, the X and Y coordinates may be
23 edited. These X and Y coordinates establish an X and Y coordinate in pixels for a
24 graphical user interface element. The Y variable displays the width of the control element
25 in pixels and the value H displays the height of the control element in pixels. The “TITLE”
26 element is a field for displaying the title of the control. The “OPTION” group cannot be

1 edited. The "TEXT" element identifies a default static control text. The "FONT" element
2 establishes a font name, size, and style. The "MULTI-LINE" element identifies an edit
3 control feature in which a multiple or single line text may be identified. The "RTF" and
4 "RTF TOOLBAR" elements identify a text box that can contain RTF text (Microsoft's rich
5 text format). The "EXTENDED" element identifies a multi-select list box that includes a
6 check box for items in that list box.

7
8 Determining the relative position of all of the elements in a graphical user interface is a
9 relatively simply matter when the operating environment and language is suitable for
10 relatively high-bandwidth communication. For example, graphical user interfaces encoded
11 in C++ or JAVA are relatively easy to implement using the present invention. However,
12 graphical user interfaces which utilize HTML are a little more complicated to implement
13 since there is no X,Y (coordinate system) in an HTML environment. In accordance with
14 one implementation of the present invention, the elements of a graphical user interface
15 may be located in HTML tables which can be utilized to emulate an X, Y coordinate
16 system and which can be utilized in real time to properly locate the user interface
17 elements. When operating an HTML environment, the present invention allows an HTML
18 page to resemble graphical interface layouts for alternative rendering engines such as
19 C++ and JAVA, utilizing a single database. Accordingly, the present invention allows a
20 single database to support multiple user interface environments so separate parallel
21 databases need not be maintained.

22
23 **Figure 14** is a pictorial representation of how elements in a graphical user interface in an
24 HTML environment are laid-out in accordance with the preferred embodiment of the
25 present invention. As is shown, HTML page 400 is divided up into segments which cover
26 the entire space that is available. In the HTML format, this is considered to be a irregular

1 table. The space within HTML page 700 will be accounted by a "cell" within the table even
2 if there is no text or image provided in that particular portion of Page 700. In the view of
3 **Figure 14**, "cells" which do not contain any text or images are shown in dashed outlined
4 form. Portions of the graphical interface which include text or images are shown in solid
5 line form. The space within HTML page 700 may be divided up into rows. The first row is
6 made up of cells 701, 703, 705, 707. Of these, cell 707 does not include any text or image
7 portions. It is merely space which is not utilized. The cell 707 is utilized by the program in
8 order to "account" for the space. The second row includes cells 709, 711, and the upper
9 portion of cell 713. Cell 711 does not contain any text or images and is utilized merely to
10 account for the space between cell 709 and cell 713. The third row is made up of cells
11 715, 717, 719, 721, and 713. Of these, cells 715, 721 are regions which do not contain
12 any text or images and which are utilized only to account for the space in that row. The
13 next row includes cells 723, 725, 727, 729, 731. Cells 725, 729 include text and/or images
14 while cells 723, 727, 731 do not include any text or images and are merely utilized to
15 account for the space in that row. This row also includes the portion of cell 713. The next
16 row is made of cells 733, 735, 737, 739, 741, 713. Of these cells, cells 733, 737, 741 are
17 not utilized for depicting any text or images and are merely utilized to account for the
18 space in that row. The next row includes cells 743, 745, 747, 749, 751 and 713. Cells
19 745, 749, 713 include text and/or images. Cells 743, 747, 751 do not include text or
20 images. The next row includes cells 753, 755, 757. Cells 755, 757 include text and/or
21 images while cell 753 does not include any text or images. The final row is made of cell
22 759.

23
24 **Figure 15** depicts HTML page 15 with cells populated with text and images in accordance
25 with the Dialog Controls 514 and Dialog Manager 512 of **Figure 12**. As is shown, cell 701,
26 703, 705 include textual material. Cell 713 includes a photo of the product. Cells 709,

1 717, 719 include textual material relating to product features. Cell 725, 735, 745, 729,
2 739, 749 include active areas which receive operator selections. Cell 755, 757 include
3 buttons. Cell 759 includes scroll back and log out text. Examination of **Figures 14 and 15**
4 reveals that the table layout of HTML page 700 is useful for managing the relative position
5 and size of the elements of the graphical user interface.

6
7 A routine which implements the preferred embodiment of the present invention is depicted
8 in the flow chart of **Figure 16**. The process starts at Block 801, and goes to Block 803,
9 wherein the product model is loaded. Next, in accordance with Block 805, the Dialog
10 Controls Module is loaded. Then, in accordance with Block 807, the Dialog Manager is
11 loaded. Next, in accordance with Block 809, the Dialog Controls Module is searched for
12 the particular product model. Once it is located, the values associated with that particular
13 product model are read from the Dialog Controls Module. Then, in accordance with Block
14 813, the size of the user interface elements are adjusted if necessary in order to have the
15 "cells" sized to an extent sufficient to carry all of the text and/or images which are to be
16 posted in that particular portion of the HTML page. This can be best understood with
17 reference again to **Figure 14**. For the third row from the top, the text resident in cells 717,
18 719 may require a greater or lesser amount of space. If these cells need to be expanded,
19 space that is not dedicated for some other purpose can be utilized. For example, cells
20 517, 521 may be further reduced in order to allow a greater amount of text or images to be
21 posted to that portion of the HTML page 700. Sizing issues may become more complex if
22 fonts sizes are changed for a particular graphical user interface. Changes in font size will
23 necessitate recalculation of the available cell space and enlargement of the cells dedicated
24 for text in order to accommodate the enlarged text.

1 Returning now to **Figure 815**, once the user interface elements are adjusted in size, then
2 the user interface is populated with its elements in accordance with Block 815 and the
3 process ends at Block 817.

4
5 One principal advantage of the present invention is that existing databases can be utilized
6 to populate on HTML pages which serve as graphical user interfaces. This can be done
7 without creating any new or separate parallel database for the same types of data. The
8 present invention can be applied to a wide variety of programming environments and
9 programming languages. The basic concept is to utilize a single data base and multiple
10 rendering engines in a controlled manner in order to enable the generation of generally
11 similar or comparable graphical user interface environments which utilize that single data
12 base to communicate with customers. In this manner, the customer may interact with a
13 seller in one programming environment/language on one occasion, but in another
14 programming environment/language on another occasion. In all locations, the graphical
15 user interface screens which are presented to the user will be generally comparable in
16 appearance. Furthermore, the data displayed on those pages will be identical since they
17 are drawn or pulled from a single database.

18
19 A feature of the present invention is that it can also be applied to multiple sellers, as well
20 as multiple buyers as described above. This would allow a flexible approach to commerce
21 for complex e-commerce systems.

22
23 Although the invention has been described with reference to a particular embodiment, this
24 description is not meant to be construed in a limiting sense. Various modifications of the
25 disclosed embodiments as well as alternative embodiments of the invention will become
26 apparent to persons skilled in the art upon reference to the description of the invention. It

1 is therefore contemplated that the appended claims will cover any such modifications or
2 embodiments that fall within the scope of the invention.

3

What is claimed is:

1. A system for communicating commercial transaction information between a Seller and a plurality of Buyers over a distributed data processing system, comprising:

(a) a single database for maintaining a plurality of user interface metadata elements including at least component identifications and component properties;

(b) a visual rule model for configuring a plurality of graphical user interface dialog pages utilizing said metadata and a plurality of dialog rules;

(c) a plurality of rendering engines each adapted to respond to commands from said visual rule model; and

(d) a dialog manager for passing at least said metadata elements to an appropriate one of said plurality of rendering engines in order to dynamically constrict a plurality of graphical user interface screens in said distributed data processing systems in order to allow the communication of information between said Seller and said plurality of Buyers necessary related to a potential commercial transaction.

2. A system according to Claim 1, wherein plurality of rendering engines include a hyper-text mark-up rendering engine.

3. A system according to Claim 1, wherein a communication connection is established between said Seller and said plurality of Buyers which is a relatively low bandwidth communication channel.

1

2

3

4

5

4. A method according to Claim 3 wherein said relatively low bandwidth communication channel comprises an internet connection.

1 **5.** A method of conducting computer-moderated commercial transactions, comprising:

2
3 (a) providing a single database which contains product metadata relating to
4 objects of computer-moderated commerce;

5
6 (b) providing a plurality of alternative rendering engines for constructing a
7 plurality of graphical user interface screens relating to said objects of computer-moderated
8 commerce;

9
10 (c) providing a dialog manager which can be utilized to provide commands to
11 said plurality of alternative rendering engines;

12
13 (d) during interaction with a customer in a computer-moderated commercial
14 transaction, utilizing said dialog manager to pass said product metadata from said single
15 database to a particular one of said plurality of alternative rendering engines to
16 dynamically construct a series of graphical user interface screens which include active and
17 passive portions for presenting a plurality of product options to said customer and to
18 record said customer's selection.

19
20 **6.** A method according to Claim 5, wherein plurality of rendering engines include a
21 hyper-text mark-up rendering engine.

22
23 **7.** A method according to Claim 5, wherein a communication connection is
24 established between said Seller and said plurality of Buyers which is a relatively low
25 bandwidth communication channel.

1 **8.** A method according to Claim 7 wherein said relatively low bandwidth
2 communication channel comprises an internet connection.

3
4 **9.** A method according to Claim 5 wherein the rendering engines comprise at least
5 three engines written in a different programming language.

1 **10.** A method of conducting a computer-moderated commercial transaction between a
2 Seller and a Buyer, comprising:

3
4 (a) providing a distributed data processing system including a relatively low-
5 bandwidth communication channel between said Seller and Buyer;

6
7 (b) providing a single database under the control of said Seller which contains
8 metadata related to the subject of said commercial transaction;

9
10 (c) providing a plurality of alternative rendering engines each of which is
11 responsive to rendering commands which is in a different programming language;

12
13 (d) providing a dialog manager program under the control of said Seller which
14 moderates the passing of metadata and rendered objects over said distributed data
15 processing system to said Buyer in the form of graphical user interface screens;

16
17 (e) passing transaction information to said Buyer over said relatively low-
18 bandwidth communication channel of said distributed data processing system in the form
19 of graphical user interface screens which confine particular relevant portions of said
20 metadata and the output of a particular appropriate one of said plurality of alternative
21 rendering engines; and

22
23 (f) receiving transaction selections from said Buyer over said relatively low-
24 bandwidth communication channel of said distributed data processing system through
25 monitoring of interaction between said Buyer and said graphical user interface screens.
26

1 **11.** A method according to Claim 10, wherein said relatively low-bandwidth
2 communication channel comprises an internet connection.

3
4 **12.** A method according to Claim 11, wherein said programming languages include a
5 hyper-text mark-up language such as HTML.
6

ABSTRACT

A system and method supports multiple alternative graphical user interfaces in computer-moderated electronic commerce. A single database system provides information necessary for standard graphical user interface display by different display engines. The information which is supplied to generate user interface objects is formatted to support low bandwidth graphical user interfaces. In particular, formatting is selected so that low bandwidth HTML engines render a user interface similar to that displayed by higher bandwidth engines.

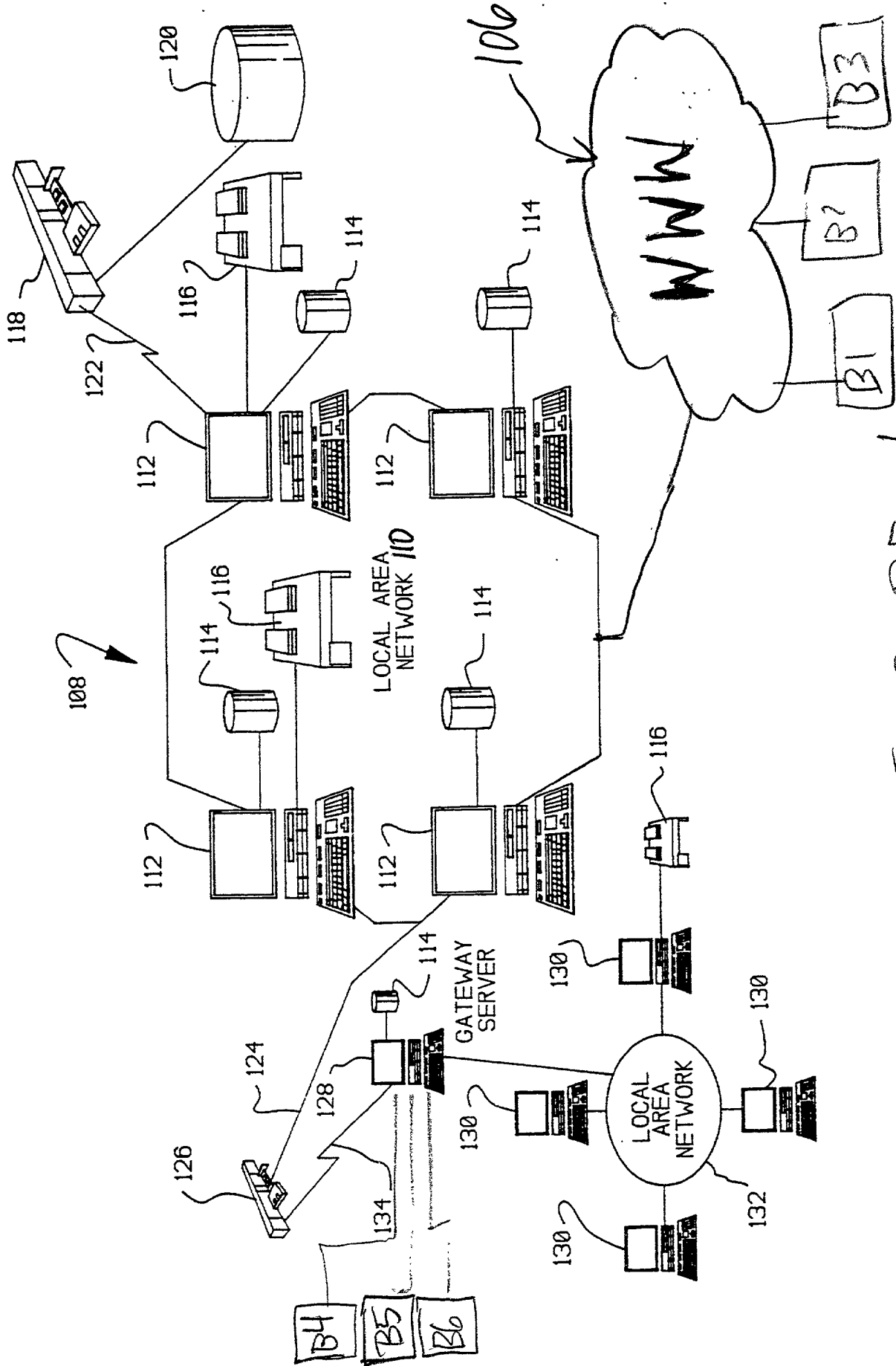
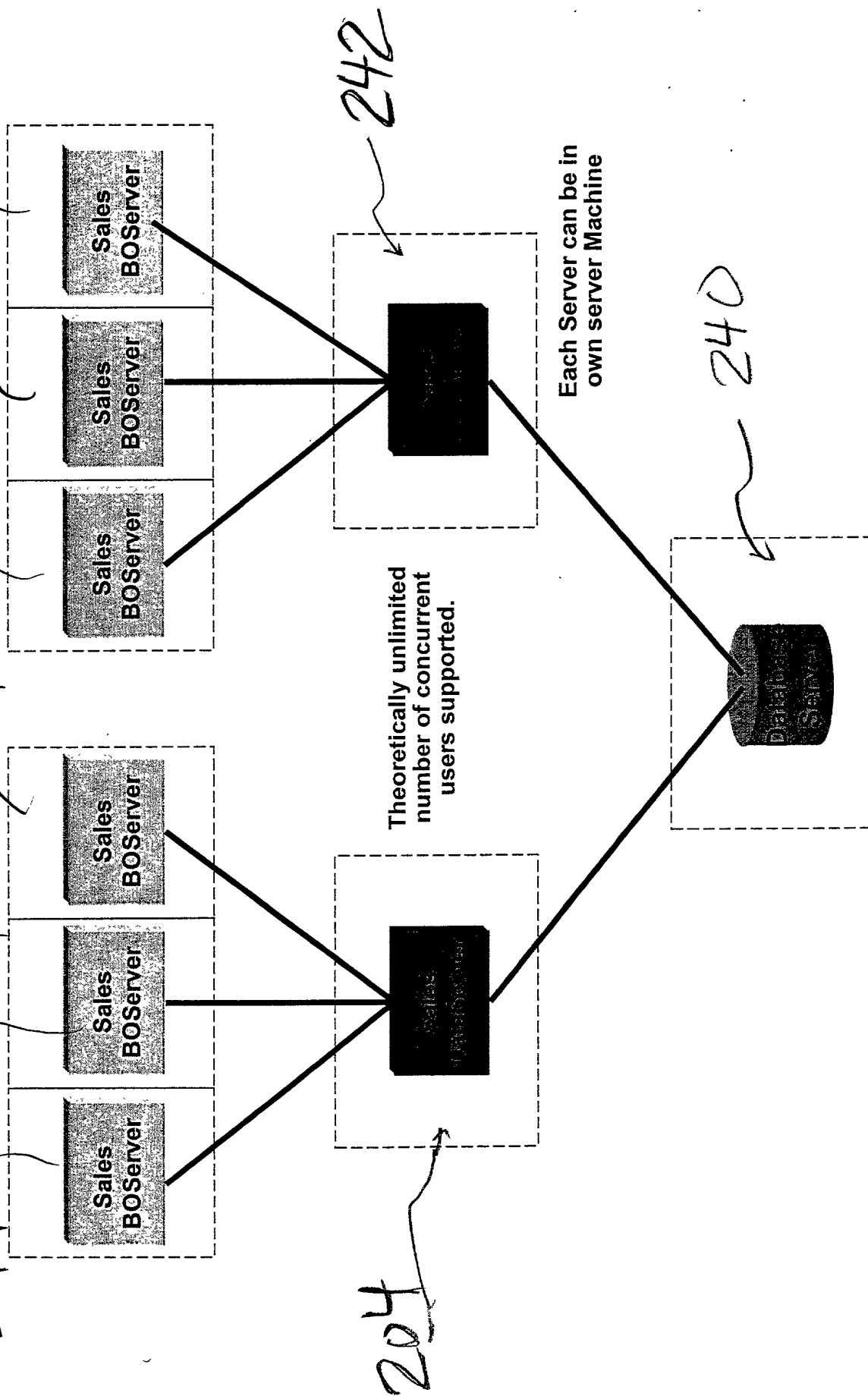


FIGURE 1

Scalability: Pyramid Model of Servers



3 JUL 1961

Scalability: Multithreaded Server

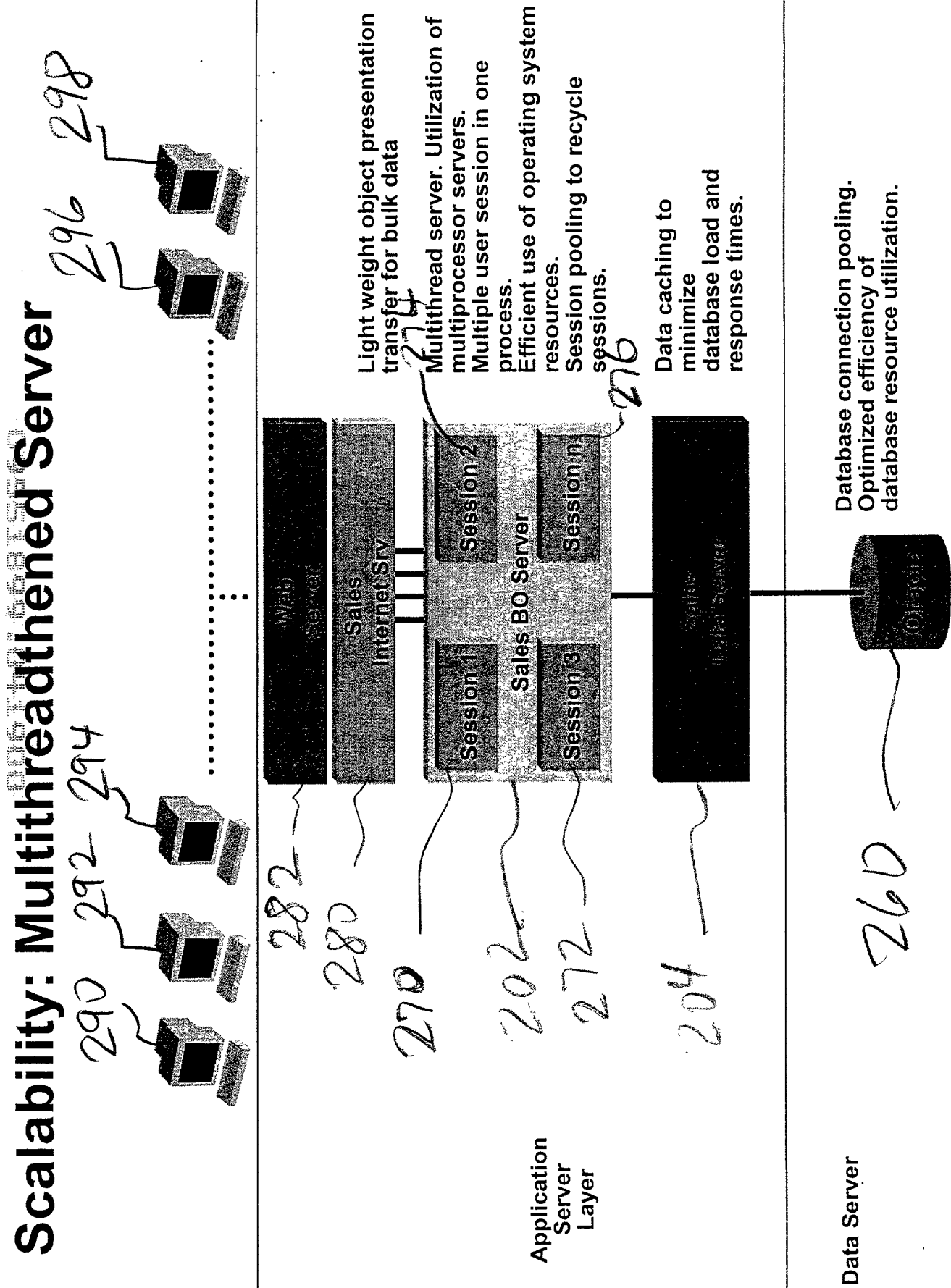


FIGURE 4

UI Summary

000T+0" 568T5560

HTML UI



STD UI



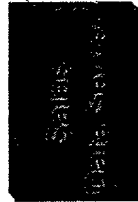
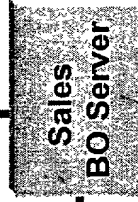
Low speed
Wan /
Internet /
Lan

HTTP

Corba IIOP

LAN / High
speed Wan

Application
Servers



Application
Servers

Data Server

Data Server

All Uis can be utilized
simultaneously using
the same data

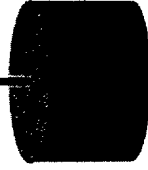


FIGURE 5

Rule Maintenance

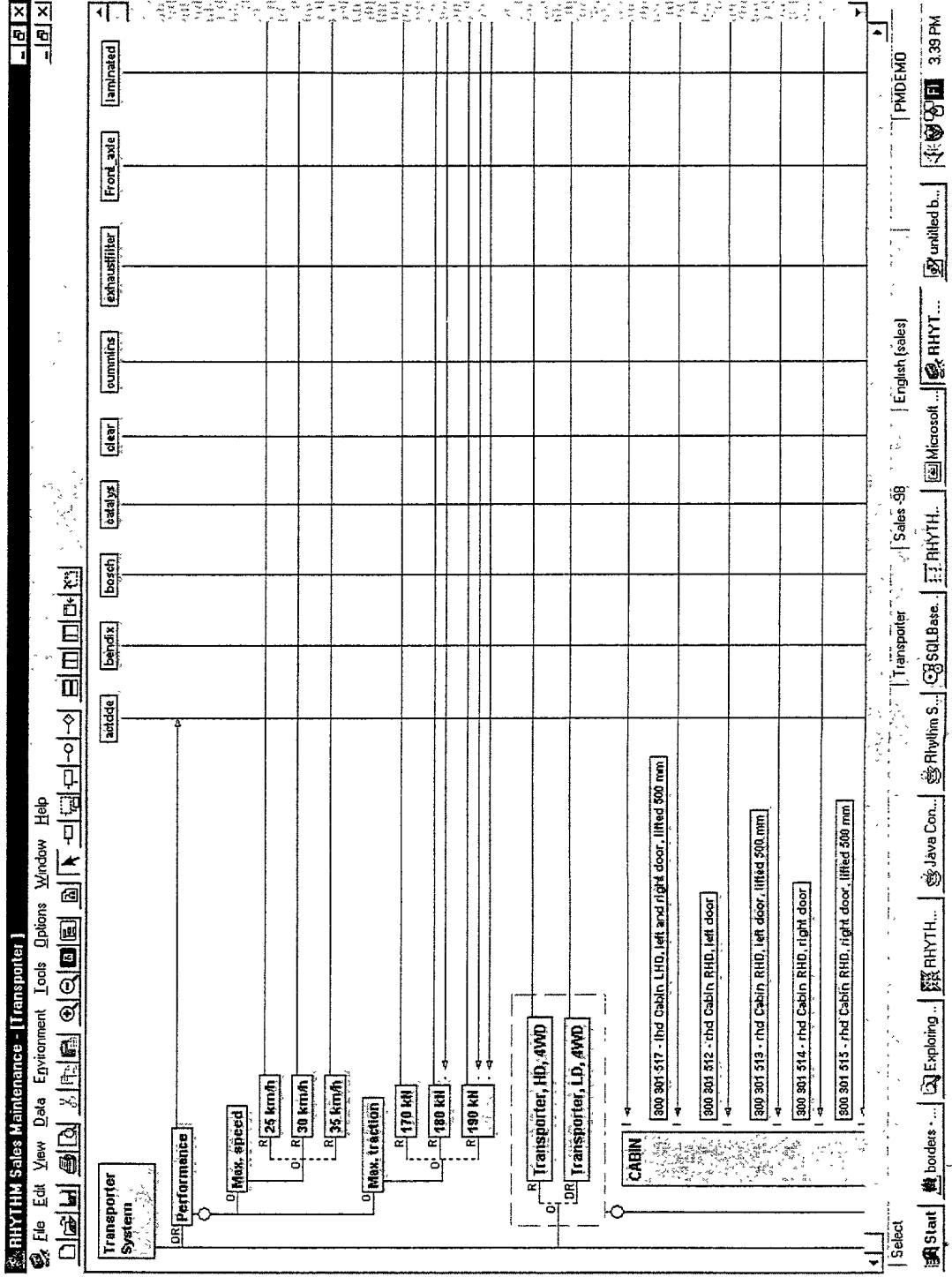


FIGURE 6

601
603

605

613
615

617

Dialog Controls

Name:

Code:

	PM-Variable	Dialog ID	Control
1	{Box.Name}	Chassis	DropListBox2

Dialog name:

Dialog ID:

Dialog Controls:

- CheckBox1
- CheckBox2
- Dialog
- ListBox1
- ListBox2
- ListBox3
- ListBox4
- PushButton1
- ToolStrip1

Buttons: OK, Cancel, <<, >>, Variables..., Clear, Add, Delete, Update, Edit..., Show..., Hide..., Help

FIGURE 7

651

Dialog Manager

ID: Name:

Controls:

- Dialog
- ListBox1
- ListBox3
- CheckBox1
- ListBox4
- ListBox2
- TextBox1
- CheckBox2
- PushButton1

Properties:

	Property	Value
1	Class	Dialog
2	X	0
3	Y	0
4	W	268
5	H	127
6	Title	Untitled

Buttons: Close, Add, Save, Delete, Dialog Editor, Help

FIGURE 8

657

UI Maintenance

Microsoft Visual Studio .NET 2003

File Edit View Data Environment Tools Options Window Help

UI Maintenance - Transporter

Dialog Manager

ID: Cabin Name: []

Controls:

- Dialog
- ListBox1
- ListBox3
- CheckBox1
- ListBox4
- ListBox2
- TextBox1
- CheckBox2
- PushButton1

Properties:

Property	Value
Class	Dialog
1	0
2	X
3	Y
4	W
5	H
6	Title
7	Font
8	HTMLPrefix
9	HTMLSuffix

Buttons: Close Add... Save Delete Exit

Dialog Editor

File Edit Controls Help

Transporter, HD, 4WD

Transporter, LD, 4WD

CABIN

300 301 517 - lhd Cabin LHD, left and right d

300 301 512 - rhd Cabin RHD, left door

300 301 513 - rhd Cabin RHD, left door, lift

300 301 514 - rhd Cabin RHD, right door

300 301 515 - rhd Cabin RHD, right door, lift

line 1 line 2 line 3 line 4 line 5

line 1 line 2 line 3 line 4 line 5

Text Box

Check Box

Push Button

laminated front axle exhaust filter pumpkins clear catalys in post

Start borders ... Exploring ... RHYTH... Java Con ... RHYTH... 3:37 PM

FIGURE 7

Abstract

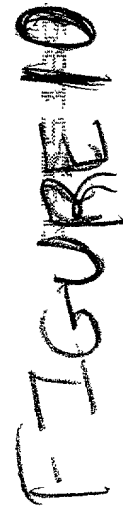


FIGURE 10

START

404

ESTABLISH COMMUNICATION BETWEEN BUYER AND SELLER

ACTIVATE APPROPRIATE RENDERING ENGINE

406

CALL APPROPRIATE DIALOG MANAGER MODULE

408

UTILIZE RENDERING ENGINE TO GENERATE ASSOCIATED GRAPHICAL USER INTERFACE

410

COMMUNICATE WITH SINGLE DATABASE TO READ META DATA ASSOCIATED WITH THE GUI

412

APPLY METADATA DURING CREATION OF GUI IN REAL TIME

414

MONITOR DIALOG INPUT BY BUYER

416

AS NECESSARY, WRITE DATA TO DATABASE

418

END SESSION

420

NO

YES

END

422

FIGURE 11

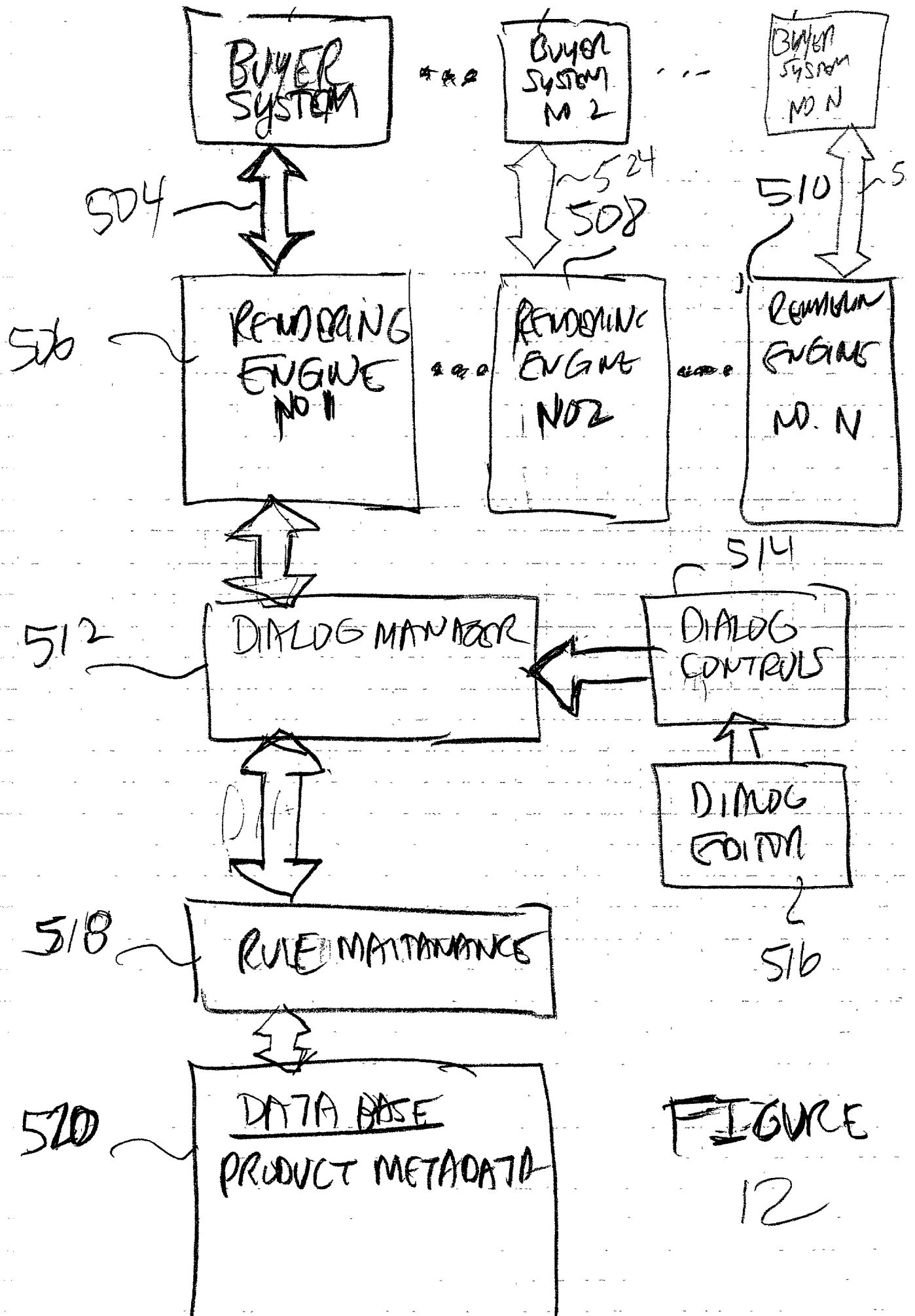
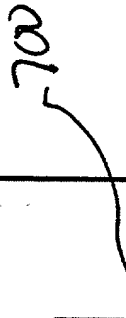


FIGURE
12

Property	Value
Class	Cannot be edited.
X	Displays the X coordinate of the control in pixels.
Y	Displays the Y coordinate of the control in pixels.
W	Displays the width of the control in pixels.
H	Displays the height of the control in pixels.
Title	Displays the title of the control.
Option Group	Cannot be edited.
Text	The default static control text
Font	Displays font name, size and style.
Multiline	Edit control feature (single line, multi line); 0 = no, 1 = yes
RTF	Defines the Edit control as RTF type; 0 = no, 1 = yes
RTFToolBar	When RTF has been chosen; a RTF toolbar can be added; 0 = no, 1 = yes
Extended	When listboxes are used; they can be owner type (can include picture, checkboxes etc.); 0 = no, 1 = yes

FIG 13

HTML table



709 715 723 731 743

FIGURE 14

CCS HTML UI

RHYTHM Sales Configuration Configure Transporter" - Netscape

File Edit View Go Communicator Help

Back Forward Reload Home Search Guide Print Security Stop

Bookmarks Location: [htm?psc_s=lbUI19203203672028&fm=G_VR_SC_LBUIServerhtml/configurator.htm&pc=1](#)

Configure "Transporter, Europe / Sales -99"

product: 701 703 705

CABIN	POWERTRAIN	CHASSIS
Performance	Max. traction	
Max. speed	170 MPH	719
25 km/h	180 MPH	725
30 km/h	190 MPH	735
35 km/h		745

[<<BACK](#) or [Logout from here!](#)

759 757 753 751

Document Done

FIGURE 15

START ~ 801

LOAD PRODUCT MODEL ~ 803

LOAD DIALOG CONTROLS MODULE ~ 805

LOAD DIALOG MANAGER ~ 807

SEARCH DIALOG CONTROLS MODULE
FOR PRODUCT MODEL ~ 809

READ VALUES FOR PRODUCT MODEL
FROM DIALOG CONTROLS MODULE ~ 811

ADJUST SIZE OF UI ELEMENTS
IF NECESSARY ~ 813

POPULATE UI ~ 815

END ~ 817

FIGURE 16

DECLARATION FOR PATENT APPLICATION

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name.

I believe that I am an original, first, and joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled

**METHOD AND APPARATUS FOR SUPPORTING MULTIPLE
ALTERNATIVE GRAPHICAL USER INTERFACES IN COMPUTER-
MODERATED ELECTRONIC COMMERCE**

further identified by attorney docket no. **0544MH-34056 (RM 143)**;

This application claims the benefit of U.S. Provisional Application Serial No. **60/130,735**, filed **20 April 1999**, entitled **Method And Apparatus For Supporting Multiple Alternative Graphical User Interfaces In Computer-Moderated Electronic Commerce**.

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose to the Office all information known to my person to be material to the patentability of this application in accordance with Title 37, Code of Federal Regulations, Sec. 1.56(a).

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

I hereby appoint **Melvin A. Hunn, Reg. No. 32,574** and **Kenneth C. Hill, Reg. No. 29,650** to prosecute this application and to transact all business in the U.S. Patent and Trademark Office in connection therewith.

006740" 66875560

Please send all correspondence to:

Kenneth C. Hill
Registration No. 29,650
Melvin A. Hunn
Registration No. 32,574
HILL & HUNN LLP
201 Main Street, Suite 1440
Fort Worth, Texas 76102
817/332.2113 (voice)

Inventor's Signature: _____

Full Name of First Joint Inventor: Harri Rajala

Date of Signature: _____

Residence and Post Office Address: _____

Citizenship: _____

Inventor's Signature: _____

Full Name of Second Joint Inventor: Sami Lahti

Date of Signature: _____

Residence and Post Office Address: _____

Citizenship: _____

005740 66072200

